Calibração Robusta de Vídeo Para Realidade Aumentada

Bruno Madeira[1,2], **Luiz Velho**[1], **Paulo Cezar Carvalho**[1]

[1]Instituto Nacional de Matemática Pura e Aplicada (IMPA)
Estrada Dona Castorina, 110, Jardim Botânico – Rio de Janeiro – RJ – Brasil

[2]Instituto Militar de Engenharia (IME)
Praça General Tibúrcio, 80, Praia Vermelha – Rio de Janeiro – RJ – Brasil

madeira@de9.ime.eb.br, {lvelho,pcezar}@visgraf.impa.br

Resumo. *Este artigo apresenta um algoritmo robusto capaz de estimar os parâmetros extrínsecos assumidos por uma câmera na captura dos quadros de um vídeo. É assumido que os parametros intrínsecos foram previamente estimados, e que estes não variam ao longo do tempo. No final do artigo é apresentado um exemplo de resultado aplicado à realidade aumentada.*

1. Introdução

Realidade Aumentada corresponde ao processo de adicionar objetos virtuais criados por computador sobre um vídeo capturado por uma câmera. Tal processo pode ser realizado em tempo real ou não. Neste artigo estamos considerando que não é exigido que o processo seja executado em tempo real, sendo aplicável, por exemplo, para criação de efeitos especiais em cinema.

Um dos problemas que precisa ser resolvido para o desenvolvimento de um sistema de realidade aumentada é o problema de calibração, que consiste na determinação dos parâmetros da câmera utilizados na captura dos quadro do vídeo que se deseja combinar com imagens sintéticas. Tais parâmetros se dividem em duas categorias: os parâmetros intrínsecos, que descrevem características da câmera como distância focal, ponto principal e resolução; e os parâmetros extrínsecos, que descrevem a posição e a orientação da câmera. Aqui estamos tratando de um algoritmo que determina os parâmetros extrínsecos de uma câmera associados aos quadros de um vídeo, assumindo-se que os parâmetros intrínsecos foram previamente estimados.

Muitos sistemas de realidade aumentada baseiam sua calibração no estabelecimento de correspondências entre pontos 3D marcados na cena, cujas coordenadas são conhecidas, sobre suas respectivas projeções 2D nos quadros do vídeo. Isto é o que ocorre, por exemplo, em aplicações desenvolvidas utilizando-se ARToolKit. Este não é o caso tratado aqui. No nosso caso, estamos considerando que não é permitido realizar nenhum tipo de marcação sobre a cena, o que torna o problema de calibração mais difícil.

O problema de calibração é resolvido pelo acompanhamento de pontos 2D dos quadros do vídeo, que são projeções de um mesmo ponto 3D da cena. Considera-se que as coordenadas 3D dos pontos que geraram tais projeções são desconhecidas. A única hipótese assumida é que não existe movimento relativo entre as superfícies da cena, ou seja, toda cena se move como um corpo rígido no vídeo.

A seleção e acompanhamento de pontos 2D do vídeo, que correspondem ao mesmo ponto 3D da cena, é feita de forma automática, pelo algoritmo Kanade-Lucas-

Tomasi (KLT) [Tomasi and Kanade 1991]. Este algoritmo não apresenta garantias de correção ou precisão, por isso, são adicionadas estratégias para aumentar sua robustez.

Muitas das idéias utilizadas em nosso algoritmo de calibração foram inspiradas em [Gibson et al. 2002]. Existem entretando grandes diferenças na estratégia empregada para aumentar a robustez do algoritmo, destacando-se o processo apresentado por nós, que chamamos de Ciclos de Refinamento, e que é o foco principal do artigo.

2. Definições

Com o objetivo de caracterizar formalmente o algoritmo de calibração, adotaremos as seguintes definições:

Câmera: Uma câmera é uma função $P : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ tal que, se $X \in \mathbb{R}^3$ é a coordenada de um ponto da cena, então $P(X)$ é sua projeção em uma imagem.

Vídeo: Um vídeo é uma família finita de imagens $(I)_n = (I_1, ..., I_n)$, onde cada imagem I_k corresponde a um quadro captado por uma câmera.

Família de pontos homólogos: Dado um vídeo $(I)_n = (I_1, ..., I_n)$, dizemos que a família $(x)_n = (x_1, ..., x_n)$, onde $x_i \in \mathbb{R}^2$, é uma família de pontos homólogos associada ao vídeo $(I)_n$ se existe um ponto $X \in \mathbb{R}^3$, da cena, tal que a projeção de X em I_j é x_j, para todo $j \in \{1, ..., n\}$.

Matriz de pontos homólogos: Uma matriz M, $m \times n$, formada por elementos de $\mathbb{R}^2$, é uma matriz de pontos homólogos associada a um vídeo $(I)_n$ se cada uma de suas linhas define uma família de pontos homólogos associada a $(I)_n$. Com essa definição temos também que a j-ésima coluna de M corresponde aos pontos homólogos do quadro I_j.

Configuração: Uma configuração é um par $((P)_n, \Omega)$, onde $(P)_n = (P_1, \ldots, P_n)$ é uma família de câmeras e $\Omega = \{X_1, \ldots, X_m\}$, com $X_i \in \mathbb{R}^3$, é um conjunto de pontos da cena.

Explicação para famílias de pontos homólogos: Estabelecida uma tolerância $\varepsilon \in \mathbb{R}^+$, definimos que uma explicação projetiva para uma família de pontos homólogos $(x)_n = (x_1, ..., x_n)$ é uma configuração $((P)_n, \Omega)$ tal que $\forall i \in \{1, ..., n\}$, $\exists X_j \in \Omega$ que satisfaz $\|P_i(X_j) - x_i\| < \varepsilon$. Para que esta definição faça sentido é necessário que a tolerância ε não seja muito grande, correspondendo no máximo ao comprimento de alguns poucos *pixels* na imagem.

Explicação para matrizes de pontos homólogos: Uma explicação projetiva para uma matriz de pontos homólogos M é uma configuração que explica todas as famílias de pontos homólogos das linhas de M.

Erro de reprojeção: Se $X \in \mathbb{R}^3$ é uma estimativa para um ponto da cena que se projeta sobre uma imagem em $x \in \mathbb{R}^2$, e P é uma estimativa para a câmera utilizada. Definimos $\|P(X) - x\|$ como sendo o erro de reprojeção de X em relação a P.

Erro de reprojeção para explicações projetivas: O erro de reprojeção associado a uma explicação projetiva $((P)_n, \Omega)$ para uma matriz de pontos homólogos M é

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|P_i(X_j) - M_{ij}\|^2,$$

onde M_{ij} é o ponto da i-ésima linha e j-ésima coluna de M.

Explicação projetiva ótima: Uma explicação projetiva para uma matriz de pontos homólogos M é ótima, se não existe outra explicação projetiva para M com erro de reprojeção inferior.

3. Formalização do problema

Este artigo descreve um algoritmo que encontra uma explicação projetiva ótima para uma matriz de pontos homólogos M, cujos elementos podem apresentar erros grosseiros. Tal consideração faz sentido, pois os elementos de M são determinados automaticamente aplicando-se o algoritmo KLT sobre o vídeo $(I)_n$, que se deseja calibrar. Os elementos de M errados precisam ser detectados de forma automática, sendo desconsiderados no computo do erro de reprojeção das explicações projetivas para M. Tem-se então que, se $((P)_n, \{X_1, ..., X_m\})$ é a resposta do algoritmo, então $(P)_n$ é a solução para o problema de calibração do vídeo $(I)_n$[1].

4. Calibração em três passos

Pode-se encontrar uma explicação projetiva $\Psi = ((P)_n, \{X_1, ..., X_m\})$ para uma matriz de pontos homólogos M pela execução dos seguintes três passos [Gibson et al. 2002]:

Passo 1: Utiliza-se as colunas de M correspondentes aos pontos homólogos de um par de quadros I_i e I_j para estimar P_i e P_j.

Passo 2: Utiliza-se o par P_i e P_j e a matriz M para estimar o conjunto $\{X_1, ..., X_m\}$.

Passo 3: Utiliza-se o conjunto $\{X_1, ..., X_m\}$ e a matriz M para estimar a família $(P)_n$.

Se os parâmetros intrínsecos da câmera são conhecidos, então, cada passo pode ser resolvido aplicando-se técnicas de álgebra linear [Hartley and Zisserman 2003].

O problema de poderem existir erros grosseiros em M pode ser resolvido combinando-se os três passos com o algoritmo *Random Sample Consensus* (RANSAC) [Fischler and Bolles 1981]. Este algoritmo permite que sejam definidos limiares para limitar o erro de reprojeção cometido na estimatição de pontos e câmeras, de forma que, linhas de M que possuam erros grosseiros sejam desconsideradas na determinação de Ψ.

5. Ciclos de refinamento

Um dos problemas de se aplicar o algoritmo RANSAC na calibração em três passos é a possibilidade de alguma família de pontos homólogos ser descartada indevidamente, devido ao fato da reconstrução tridimensional realizada pelo passo 2 não apresentar boa precisão, por ser calculada a partir de projeções obtidas por um único par de câmeras. Resolvemos este problema desenvolvendo um algoritmo criado a partir de uma modificação do algoritmo de calibração feito com Levenbeg-Marquardt [Hartley and Zisserman 2003]. Conseguimos, dessa forma, selecionar de maneira mais criteriosa as famílias de pontos homólogos que precisam ser efetivamente desconsideradas.

Inicialmente é determinada uma explicação projetiva $((P)_n, \Omega_1)$ obtida pela calibração em três passos. Esta solução é então refinada pela execução de um algoritmo formado por ciclos de quatro passos:

1. Executam-se algumas iterações do algoritmo Levenbeg-Marquardt, utilizando como estimativa inicial a explicação projetiva $((P)_n, \Omega_1)$, determinando-se uma outra explicação projetiva $((P')_n, \Omega_2)$ de menor erro de reprojeção associado.

[1]Em geral, não é possível associar uma matriz de pontos homólogos a um vídeo muito longo. Neste caso, o que se faz, é fragmentar o vídeo em diversos vídeos menores que são calibrados isoladamente. Posteriormente, faz-se a junção das famílias de câmeras associada aos fragmentos, obtendo-se a calibração do vídeo original.

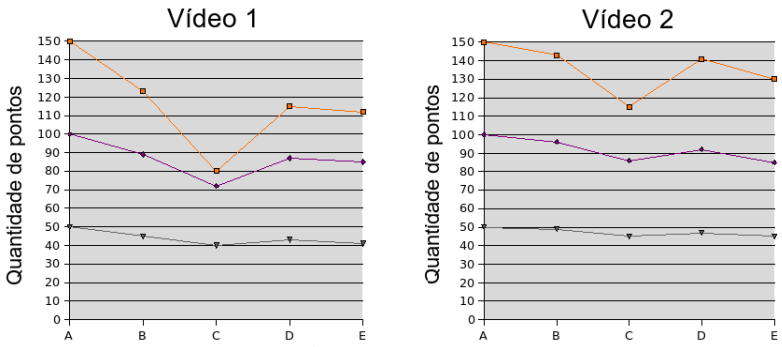

Figura 1. (A) Pontos selecionados pelo KLT no início do fragmento; (B) Pontos acompanhados pelo KLT por todo o fragmento; (C) Pontos que não foram elininados pelo RANSAC durante a calibração em três passos; (D) Pontos reconstruídos pelo primeiro ciclo de refinamento; (E) Pontos reconstruídos pelo segundo ciclo de refinamento.

2. Utilizam-se pares de câmeras de $(P')_n$ para determinar uma nova reconstrução Ω_3 para todos os pontos homólogos de M. Esse processo pode ser realizado escolhendo-se pares de câmeras diferentes para reconstruir cada ponto de Ω_3, de forma, que cada par utilizado seja aquele que minimiza o erro de reprojeção associado a cada ponto.
3. Descartam-se os pontos de Ω_3 cujos erros de reprojeção em relação à alguma das câmeras de $(P')_n$ são maiores que um limiar $\xi \in \mathbb{R}^+$. Obtém-se assim um novo conjunto de pontos Ω_4.
4. Estima-se uma nova família de câmeras $(P'')_n$ a partir do conjunto de pontos Ω_4 e das respectivas linhas da matriz de pontos homólogos M. Com isso, obtemos uma explicação projetiva $((P'')_n, \Omega_4)$, que pode ser utilizada para alimentar um novo ciclo de refinamento.

A cada ciclo pode-se utilizar um limiar ξ menor, tendo em vista que, como a solução fica cada vez mais correta, podemos ser cada vez mais rigorosos.

Destacamos que Ω_3 é determinado utilizando-se todas as linhas de M. Como conseqüência, tem-se que $\#\Omega_3 \geqslant \#\Omega_1$. É esse fato que possibilita, que pontos descartados indevidamente durante a calibração em três passos, possam ser readmitidos durante a execução dos ciclos de refinamento. Ou seja, torna possível que se tenha $\#\Omega_4 > \#\Omega_1$.

Após terem sido executados um determinado número de ciclos de refinamento pode-se aplicar o algoritmo Levenberg-Marquardt até sua convergência, obtendo uma explicação projetiva, cujo erro de reprojeção associado às famílias de pontos homólogos selecionadas é um mínimo local.

6. Resultados

A Figura 1 apresenta dois gráficos que indicam a quantidade de pontos utilizada nas diversas etapas da calibração de dois fragmentos de vídeos diferentes, com duração aproximada de dois segundos. Cada gráfico exibe três curvas, que correspondem aos resultados associados a seleções de 50, 100 e 150 pontos, pelo KLT, no primeiro quadro do fragmento.

O limiar de aceitação para o erro de reprojeção estabelecido para o RANSAC durante a execução do algoritmo de calibração em três passos foi de 5 pixels. Após o término

Figura 2. Quadros de um vídeo em que foi aplicado o algoritmo apresentado neste artigo. Os pontos marcados nas imagens foram escolhidos e acompanhados automaticamente pelo algoritmo KLT, sendo utilizados pelo processo de calibração, que estimou as câmeras empregadas na visualização do cubo.

deste algoritmo foram executados dois ciclos de refinamento, o primeiro utilizando um limiar $\xi = 3$ pixels, e um segundo utilizando um limiar $\xi = 2$ pixels.

Os gráficos deixam claro que, a combinação de RANSAC com ciclos de refinamento permite um melhor aproveitamento dos pontos acompanhados pelo KLT do que o uso exclusivo de RANSAC. Basta observar que, a quantidade de pontos satisfazendo o limiar de 3 pixels, adotado em (D), foi sempre maior do que a dos pontos que satisfizeram o limiar de 5 pixels, adotado em (C), e, em muitos casos, a quantidade satisfezendo o limiar de 2 pixels, em (E), também superou (C). Além disso, se no lugar deste processo combinado fosse aplicado isoladamente um RANSAC com tolerância de 2, ou 3 pixels, a quantidade de pontos descartada indevidamente seria ainda maior que a ocorrida em (C).

7. Conclusão

Apresentamos um novo algoritmo que adiciona robustez ao acompanhamento automático de pontos em um vídeo, no contexto de calibração de câmeras. Em vez de se utilizar um RANSAC muito restritivo durante a calibração em três passos, utilizou-se um RANSAC mais tolerante seguido por ciclos de refinamento que se tornam gradativamente mais restritivos. Dessa forma, o descarte indevido de pontos bem acompanhados foi reduzido.

Foi desenvolvido um protótipo em que o algoritmo foi empregado em realidade aumentada, como ilustrado na Figura 2. Na versão atual não hove preocupação com performance, sendo este um assunto deixado para um trabalho futuro.

A versão completa deste trabalho encontra-se em *www.visgraf.impa.br/ar/ar.pdf*.

Referências

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

Gibson, S., Cook, J., Howard, T., Hubbold, R., and Oram, D. (2002). Accurate camera calibration for off-line, video-based augmented reality. In *International Symposium on Mixed and Augmented Reality (ISMAR'02)*, page 37.

Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in computer vision, second edition*. Cambrige University Press, Cambridge, United Kingdom.

Tomasi, C. and Kanade, T. (1991). Detection and tracking of point features. *Technical Report CMU-CS-91-132*, 24(6).